Amendments to the Claims

Kindly amend claims 1, 7, 9, 15, 17 & 23, and cancel claims 6, 14 & 22 (without prejudice), as set forth below. All pending claims are reproduced below, with changes in the amended claims shown by underlining (for added matter) and strikethrough/double brackets (for deleted matter).

1. (Currently Amended) A computer system comprising:

tasks potentially contending for a latch, each task comprising:

a probability determining component to dynamically estimate the probability that the task will successfully acquire the latch; and

a suspending component to place the task in a suspended state for a defined sleep time where the estimated probability is below a predetermined threshold value, and wherein the suspending component bases the defined sleep time on a predicted number of instructions executed under the latch as calculated by a sample workload measurement.

2. The computer system of claim 1 in which the suspending component increments the defined sleep time by a heuristically determined constant factor for successive entries of the task into the suspended state.

3. The computer system of claim 2 in which the sleep time is capped at a predetermined maximum value.

4. The computer system of claim 1 in which the suspending component adjusts the defined sleep time in accordance with changes in the estimated probability that the task will successfully acquire the latch.

5. The computer system of claim 4 in which the sleep time is capped at a predetermined maximum value.

6. (Canceled).

7. (Currently Amended) The computer system of claim [[6]] 1, in which the sleep time is capped at a predetermined maximum value.

8. The computer system of claim 1 in which the probability determining component estimates the probability that the task will successfully acquire the latch by taking the inverse of the number of tasks contending for the latch.

9. (Currently Amended) A method for the management of contention for a latch by a task in a multitask computer system, the method comprising:

a. the task dynamically estimating the probability that the task will successfully acquire the latch;

b. the task placing itself in a suspended state for a defined sleep time where the estimated probability is below a predetermined threshold value; and

c. the task repeating the above a and b until the dynamically estimated probability of the task acquiring the latch is at or above the predetermined threshold value, following which the task will contend for the latch; and

wherein the method further comprises determining the defined sleep time by a predicted number of instructions executed under the latch as calculated by a sample workload measurement.

10. The method of claim 9, further comprising incrementing the defined sleep time by a heuristically determined constant factor for successive entries of the task into the suspended state.

11. The method of claim 10, further comprising capping the defined sleep time at a predetermined maximum value.

12. The method of claim 9, further comprising adjusting the defined sleep time in accordance with changes in the estimated probability that the task will successfully acquire the latch.

13. The method of claim 12, further comprising capping the defined sleep time at a predetermined maximum value.

14. (Canceled).

15. (Currently Amended) The method of claim [[14]] 9, further comprising capping the defined sleep time at a predetermined maximum value.

16. The method of claim 9, wherein the estimating the probability that the task will successfully acquire the latch comprises taking the inverse of the number of tasks contending for the latch to define the probability.

17. (Currently Amended) A program storage device readable by a multitasking machine, tangibly embodying a program of instructions executable by the machine to perform a method for the management of contention for a latch by a task in a multitask computer system, the method comprising:

a. the task dynamically estimating the probability that the task will successfully acquire the latch;

b. the task placing itself in a suspended state for a defined sleep time where the estimated probability is below a predetermined threshold value; and

c. the task repeating the above a and b until the dynamically estimated probability of the task acquiring the latch is at or above the predetermined threshold value, following which the task will contend for the latch; and

wherein the method further comprises determining the defined sleep time by a predicted number of instructions executed under the latch as calculated by a sample workload measurement.

18. The at least one program storage device of claim 17, wherein the method further comprises incrementing the defined sleep time by a heuristically determined constant factor for successive entries of the task into the suspended state.

19. The at least one program storage device of claim 18, wherein the method further comprises capping the defined sleep time at a predetermined maximum value.

20. The at least one program storage device of claim 17, wherein the method further comprises adjusting the defined sleep time in accordance with changes in the estimated probability that the task will successfully acquire the latch.

21. The at least one program storage device of claim 20, wherein the method further comprises capping the defined sleep time at a predetermined maximum value.

22. (Canceled).

23. (Currently Amended) The at least one program storage device of claim [[22]] 17, wherein the method further comprises capping the defined sleep time at a predetermined maximum value.

24. The at least one program storage device of claim 17, wherein the estimating the probability that the task will successfully acquire the latch comprises taking the inverse of the number of tasks contending for the latch to define the probability.

* * * * *